

Tecnología Raspberry Pi

Patricia Guadalupe Flores Bautista FB21010
Universidad de El Salvador
Facultad de Ingeniería y Arquitectura
Escuela de Ingeniería de Sistemas Informáticos

Resumen – Este documento se enfoca en estudiar, analizar y ofrecer una perspectiva detallada de la tecnología Raspberry pi, la cual ha revolucionado la informática al emerger como una plataforma computacional versátil. Esta investigación permite explorar los aspectos fundamentales de la tecnología Raspberry pi, realizando una comparación con Arduino y de esta manera identificar tanto sus diferencias como las similitudes entre ambas. Además, se presentan tres proyectos prácticos que ilustran el potencial creativo de Raspberry pi en aplicaciones del mundo real. En conjunto, este trabajo es una visión completa de las capacidades y aplicaciones de esta tecnología.

I. INTRODUCCIÓN

Este documento aborda la tecnología Raspberry Pi desde sus características fundamentales hasta su aplicación en diversos proyectos. La Raspberry Pi es un ordenador de placa unidad SBC que ha revolucionado a lo largo del tiempo en el mundo de la informática este es de bajo costo y de código abierto desde que fue lanzado en 2012. El dispositivo es compacto y versátil capturando la atención de educadores y profesionales debido a la capacidad de ejecutar un sistema operativo completo.

La investigación se enfoca en explorar los aspectos esenciales que diferencian la Raspberry Pi de dispositivos similares como el caso del Arduino, destacando capacidad, conectividad y versatilidad en aplicaciones. A lo largo de la investigación se describen tres proyectos concretos que demuestran el potencial de la tecnología Raspberry Pi.

Además, se profundiza en los modelos de las primeras iteraciones hasta las más recientes Raspberry Pi, analizando sus características, mejoras en cada modelo y aplicaciones específicas, por otro lado, se hace una comparación entre Raspberry Pi y Arduino resaltando diferencias fundamentales en cada placa, así como el uso práctico.

La investigación tiene como objetivo proporcionar una visión clara de ambas tecnologías Raspberry Pi y Arduino, iniciando en la conceptualización, implementación, experimentación e innovación.

II. OBJETIVOS

A. *Objetivo General*

1. Explorar exhaustivamente la tecnología Raspberry pi, comprendiendo aspectos fundamentales que la diferencian. Además, se busca demostrar su versatilidad mediante proyectos prácticos.

B. *Objetivos Específicos*

1. Describir de manera concisa las características y funcionalidades de la tecnología Raspberry pi.
2. Identificar las principales diferencias entre Raspberry pi y la Arduino basándonos en términos de diseño, capacidades y aplicaciones.
3. Seleccionar y presentar tres proyectos prácticos reales que demuestren el potencial y la versatilidad de la tecnología Raspberry Pi.

III. RESULTADO DE LA INVESTIGACIÓN

A. *Investigación General*

¿Qué es Raspberry pi?

Raspberry Pi es un pequeño o mini ordenador que se presenta en una sola placa base, es de bajo costo y con tamaño compacto, puede ser conectada a un monitor de computador o una tv y usarse con un mouse y un teclado estándar. Es un pequeño computador que corre en un sistema operativo Linux permitiendo a las personas explorar la computación y aprender a programar en lenguajes como *Scratch* y *Python*. Es capaz de realizar la mayoría de las tareas típicas de un computador de escritorio desde navegar en internet, reproducir videos en alta resolución, manipular documentos de ofimática, hasta reproducir juegos.

Raspberry pi puede ser usada en una amplia variedad de proyectos digitales como: reproductores de música y video, detectores, estaciones meteorológicas hasta cajas de aves con cámaras infrarrojas.

Raspberry pi fue creada en 2010 por Raspberry Pi Foundation, lanzaron dos modelos, es decir el modelo A y el modelo B.

Sumando los pines anteriores son un total de 12 de 40 pines, los 28 pines restantes son pines GPIO, en la imagen pinout se pueden observar claramente representados por las letras GPIO

Raspberry Pi

Función: Se diseña principalmente para enseñar informática en las aulas.

Interfaz entradas y salidas: Posee múltiples puertos USB, HDMI, Ethernet y GPIO, permitiendo una conexión versátil.

Programación: Tiene una variedad de lenguajes para programar entre estas están Java, Python, C/C++.

Almacenamiento/Conectividad: Incorpora su almacenamiento en forma de tarjeta microSD soportando conexiones Wifi y Bluetooth permitiendo conectividad inalámbrica.

Costo: Es mucho más costoso debido a su potencia y capacidades de computación.

Arduino

Función: Se utiliza para crear proyectos eléctricos y prototipos utilizando como filosofía aprendes a través de la experimentación.

Interfaz entradas y salidas: Sus E/S están equipadas como digitales y analógicas ideales para interactuar con sensores o actuadores.

Programación: Utiliza un IDE específico para programar en lenguaje C/C++, optimizado para proyectos embebidos.

Almacenamiento/Conectividad: No posee almacenamiento, la programación y los datos se almacenan en la memoria flash del microcontrolador. Su conectividad dependerá del modelo o módulos

adicionales.

Costo: Es mucho más económico diseñado para aplicaciones de control de monitoreo en tiempo real.

C. Proyectos usando Raspberry pi

Proyecto 1

Barra de progreso de Workday con integración de Google Calendar



Fig. 4. Calendario LED

Descripción

El proyecto del Calendario de Tiras LED para Raspberry Pi Pico W de código abierto que utiliza una tira de luces LED para visualizar de manera atractiva y práctica la jornada laboral. Esta solución innovadora integra el microcontrolador Raspberry Pi Pico W con Google Calendar, permitiendo que los eventos del día se representen en colores diferenciados en la tira de luces.

La principal funcionalidad de este proyecto es proporcionar una representación visual de los eventos programados durante el día, desde el inicio hasta el final, ofreciendo una manera elegante de mantenerse informado sobre la agenda laboral. Los usuarios pueden conectar el sistema a su calendario de Google para que las franjas horarias reservadas se iluminen con colores específicos, facilitando la visualización de los compromisos diarios.

Objetivo

Desarrollar un sistema interactivo que utilice una tira LED para visualizar de manera atractiva todos los eventos que estén contemplados en la lista del calendario de Google.

Lista de materiales y su costo

TABLA 8
MATERIALES Y COSTOS

Material	Costo
Laptop	\$ 500.00
Cortadores de alambre	\$ 7.00
Cable USB – C a USB – A	\$ 5.00
para programar la Pi	
Raspberry Pi Pico W	\$ 6.00
Tira LED direccionable de 5v	4 15.00
Perfil LED de aluminio	\$ 15.00
Cables	\$ 5.00

Carcasa plástica para la Pi \$ 9.00



Fig. 5. Materiales Calendario LED

Guía de montaje y/o programación.

Paso 1. Preparación de la Raspberry Pi Pico W

Paso 1.1. Instalar el firmware más reciente en la placa.

Paso 1.2. Descargar Thony, una herramienta accesible para editar y subir código, esta es compatible con SO Linux, Windows y Mac.

Paso 1.3. Adquirir el código del proyecto de GitHub a continuación se muestran los pasos:

```
cd ~
git clone https://github.com/veebch/hometime.git
cd hometime
git checkout a7f061caabb1aeede6bdeb07b33bb97bfbb6a420
mv config_example.py config.py
```

Paso 2. Modifique el config.py de acuerdo a sus requisitos.

Paso 2.1 Establezca su zona horaria "TIMEZONE"

Paso 2.2. Elija en número de pixeles LED "PIXELS"

Paso 2.3. Configure el "GPIOIN" para controlar la tira de LED

Paso 2.4. Configure la tira para que muestre de derecha a izquierda "FLIP"

Paso 2.5. Cambie "SCHEDULE" en el diccionario de configuración

Paso 3. Sube el código

Paso 3.1 Conecte el Pico W a su computadora laptop

Paso 3.2 En Linux puede usar `ls` luego buscar `/dev/ttyCM0` o `/dev/ttyUSB0`, tal como se ve en los siguientes comandos:

```
ls -l /dev/tty*
....
crw-rw---- 1 root dialout 166, 0 iul 8 11:44
/dev/ttyACM0
```

Paso 3.3. Abrir el IDE Thony y navegue hasta la ubicación donde clono el repositorio.

Paso 3.4 Seleccione 'ap_templates', 'uf', 'config.py', 'main.py' usando Ctrl, luego haga clic derecho en la selección y elija "Cargar en"/.

Paso 3.5. Para iniciar el programa en modo de depuración, simplemente haga clic en "main.py" en la sección "Raspberry Pi Pico" en la parte inferior izquierda. A continuación, debería ver algo similar a lo siguiente en el registro:


```
MicroPython v1.20.0 on 2023-04-26; Raspberry Pi Pico
W with RP2040
Type "help()" for more information.
>>> %Run -c $EDITOR_CONTENT
Entering setup mode...
2023-07-08 12:05:52 [info / 152kB] > starting catch
all dns server on port 53
2023-07-08 12:05:52 [info / 149kB] > starting web
server on port 80
run for 18000
```

debería ver algo como:

```
023-07-08 16:01:50 [debug / 125kB] > parsed template:
ap_templates/redirect.html (took 8 ms)
2023-07-08 16:01:51 [info / 123kB] > POST /chat (200 OK) [133ms]
Saving wifi credentials...
2023-07-08 16:02:00 [debug / 141kB] > parsed template:
ap_templates/configured.html (took 10 ms)
2023-07-08 16:02:00 [info / 139kB] > POST /configure (200 OK)
[418ms]
2023-07-08 16:02:00 [debug / 129kB] > parsed template:
ap_templates/redirect.html (took 9 ms)
2023-07-08 16:02:00 [info / 126kB] > GET /generate_204 (200 OK)
[222ms]
2023-07-08 16:02:00 [debug / 149kB] > parsed template:
ap_templates/redirect.html (took 17 ms)
2023-07-08 16:02:00 [info / 147kB] > GET /generate_204 (200 OK)
[227ms]
Resetting...
Connection lost (device reports readiness to read but returned no
data (device disconnected or multiple access on port?))

Use Stop/Restart to reconnect.
```

ver el Pico conectado a la red Wi-Fi. En esta etapa, algo como esto debería mostrarse en la serie. Al mismo tiempo, la tira LED se iniciará con un patrón de arco iris antes de mostrar la barra de progreso.

```
MicroPython v1.20.0 on 2023-04-26; Raspberry Pi Pico W with
RP2040
Type "help()" for more information.
>>> %Run -c $EDITOR_CONTENT
2023-07-08 17:24:27 [debug / 151kB] - got ip address
Connected to wifi, IP address 192.168.1.110
Entering application mode.
Connected to WiFi
Rainbow!
Turn off all LEDs
Grab time:
https://timeapi.io/api/TimeZone/zone?timezone=Europe/Buchares
t

setting time
Begin endless loop
Updating from Google Calendar
['2023-07-07T09:00:00+03:00', '2023-07-07T09:15:00+03:00',
'2023-07-08T09:00:00+03:00', '2023-07-08T09:30:00+03:00',
'2023-07-08T19:00:00+03:00', '2023-07-08T19:30:00+03:00']
['2023-07-07T09:00:00+03:00', '2023-07-07T09:15:00+03:00',
'2023-07-08T09:00:00+03:00', '2023-07-08T09:30:00+03:00',
'2023-07-08T19:00:00+03:00', '2023-07-08T19:30:00+03:00']
clockin: 9.0
True 9.0 19.5 17.41333
Pour yourself a cup of ambition
...
```

navegador.

Paso 4.2. Abre Google Calendar:

<https://calendar.google.com/calendar>

Paso 4.3. Haz clic en "Otros calendarios" y luego en "Crear nuevo calendario".

Paso 4.4 Establezca un nombre y una descripción para su nuevo calendario, luego haga clic en "Crear calendario".

Paso 4.5. Haga clic en los tres puntos junto al nombre del calendario en la barra izquierda, luego seleccione "Configuración y uso compartido".

Paso 4.6. Marca la casilla "Poner a disposición del público".

Paso 4.7 Desplácese hacia abajo y copie el "ID de calendario" para usarlo más tarde.

Paso 4.8. Abrir

<https://console.cloud.google.com/projectselector2/apis/dashboard>

Paso 4.8. Haz clic en "Crear proyecto", establece un nombre para el proyecto y haz clic en "Crear".

Paso 4.9. En la página del proyecto, busca la API de "calendario", haz clic en "API de Google Calendar" y luego haz clic en "Habilitar".

Paso 4.10. Vaya a "Credenciales", seleccione "Crear credenciales", luego "Clave de API" y copie esta clave para usarla más adelante.

Paso 4.11. Vuelve a la página de Google Calendar <https://calendar.google.com/calendar> y añade dos eventos recurrentes para cada día: un evento de "día de inicio" de 9:00 a 9:05 y un evento de "día de finalización" de 17:00 a 17:05.

Paso 4.12. Pruebe la integración mediante comandos curl en un terminal Linux. Reemplace los marcadores de posición en las primeras líneas con sus valores reales:

```
API_KEY="YOUR_API_KEY"
TIME_MIN="2023-05-21T00:00:00Z"
TIME_MAX="2023-05-28T23:59:59Z"
CALENDAR_NAME="YOUR_CALENDAR_ID"
encoded_calendar_name=$(echo "$CALENDAR_NAME" | sed 's/
/%20/g')

curl
"https://www.googleapis.com/calendar/v3/calendars/${encoded_calen
dar_name}/events?key=${API_KEY}&timeMin=${TIME_MIN}&timeMax=
${TIME_MAX}"
```

Debería devolver un JSON como este:

```
{
  "kind": "calendar#events",
  "etag": "\"p33sff2nriuvu0g\"",
  "summary": "Hometime",
  "updated": "2023-07-05T08:50:54.763Z",
  "timeZone": "Europe/Bucharest",
  "accessRole": "freeBusyReader",
  "defaultReminders": [],
  "nextSyncToken": "CPj3ivuX9_8CEAAYCiDR6oaDAg==",
  "items": [
    {
      "kind": "calendar#event",
      "etag": "\"3377094035406000\"",
      "id": "7mo8219d0ggad882ua6t3gcash_20230705T060000Z",
      "status": "confirmed",
      "htmlLink":
"https://www.google.com/calendar/event?eid=N21vODJsOWQwZ2dhZDg4MnVhNnQzZ2Nhc2
hfMjAyMzA3MDVUMDYwMDAwWiBjX2QzODM0YTc4YzQ0NDU3OTIyMTBkNWwMTTRlNjRjMTAzNzRkNDU
xYzQxMzdkM2EwNTFjMGU5MjQzMmFjNTQ1OWJAZWw="
    }
  ]
}
```

Paso 4.13. Modifique las siguientes líneas de config.py y cárguelas a través de la pi

```
CALENDAR = "your calendar id"
APIKEY = "INSERT YOUR APIKEY"
...
GOOGLECALBOOL = True # ensure this is set to True
...
```

Paso 5. Cableado y creación del dispositivo

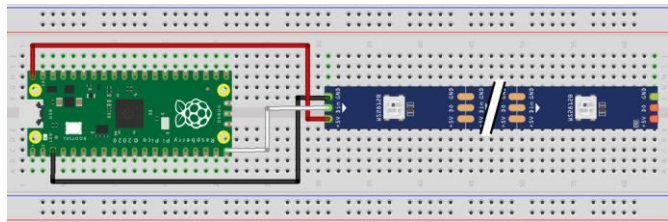


Fig. 6. Estructura del cableado



Fig. 7. Cableado final

Paso 5.1. Conecte el cable USB del cargador al Pico

Paso 5.2. A continuación, debe identificar los pines de la tira de LED.

Paso 5.3. Conecte el VBUS de la Pi a la tira de LED (+).

Paso 5.4. Conecte una de las Pi (-) a la tira de LED (-).

Paso 5.5. Conecte el pin 15 de la Pi al pin de señal de la tira de LED. Tenga en cuenta que el pin GPIO 15 es el pin número 20. Por favor, consulte en la siguiente imagen de pines.

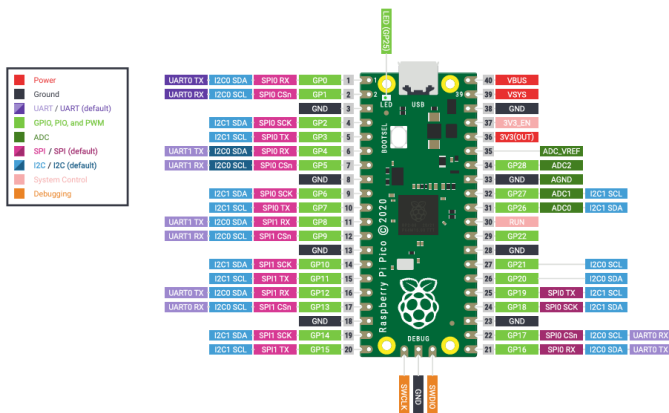


Fig. 8. Pinout entradas y salidas

Paso 5.6. En este punto, la tira de LED debería encenderse (siempre que esté dentro de los intervalos de hora correctos).

Mide el consumo de energía con un multímetro y es inferior a 500 mAh. Tenga en cuenta que el VBUS no pasa a través del Pico, sino que se extrae directamente de su cargador USB. Asegúrese de que el cargador pueda entregar más de 500 mAh para estar seguro.

Luego, soldé algunos cables más largos y delgados del Pico a la tira de LED y coloqué el Pico dentro de una carcasa de plástico. A continuación, fijé el perfil LED de aluminio con la tira y enmascaré toda la configuración.

Al final tendría que verse como en la imagen “Calendario LED”, ver [7].

Proyecto 2

Rastreador de la Estación Espacial Internacional



Fig. 9. Rastreador ProtoStax ISS

Descripción

El rastreador de la Estación Espacial Internacional (ISS) emplea la tecnología de la Raspberry Pi para acceder a los datos de ubicación en tiempo real de la ISS a través de servicios en línea. Estos datos se procesan y se muestra la información relevante en una pantalla de papel electrónico, ofreciendo una representación visual nítida y de bajo consumo energético.

La carcasa ProtoStax proporciona un entorno seguro y adecuado para la Raspberry Pi, asegurando un montaje robusto y compacto del sistema. La integración de Python permite la captura, procesamiento y visualización eficiente de los datos de ubicación, lo que resulta en una experiencia interactiva y educativa para los entusiastas del espacio y la tecnología.

Objetivo

Desarrollar un sistema utilizando Raspberry Pi, una pantalla de papel electrónico y Python, que permita rastrear en tiempo real la ubicación y trayectoria de la Estación Espacial Internacional (ISS), brindando una representación visual clara y educativa para entusiastas del espacio y la tecnología.

Lista de materiales y su costo

TABLA 9 MATERIALES Y COSTOS		
Material		Costo
Raspberry Pi 4 Modelo B		\$ 80.00
Caja ProtoStax para		\$ 7.00
Raspberry Pi B+ / Modelo 4B		
Waveshare 2.7inch e-Paper HAT (B)		\$ 18.99

SO Raspberry Pi Raspbian

Guía de montaje y/o programación.

Paso 1. Configurar la Raspberry Pi, habilitando la SPI.

Paso 1.1 Iniciar raspi-config.

Vaya a "Opciones de interfaz". Vaya a SPI y pulse Intro. Diga que sí cuando se le pregunte si desea que se habilite la interfaz SPI.

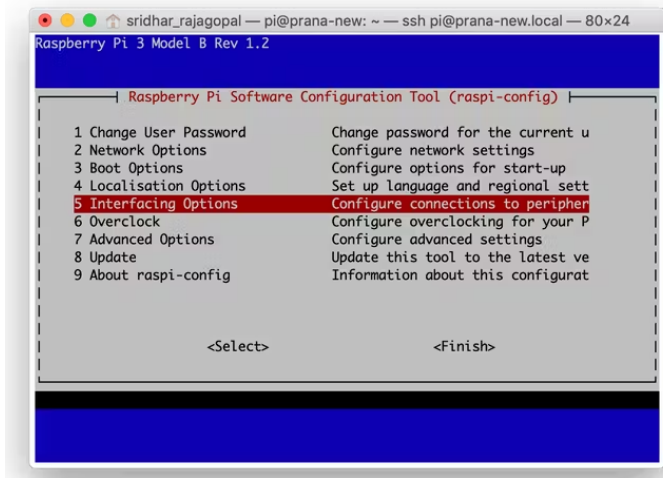


Fig. 10. Configuración SPI

Paso 1.2 Instale el código de ProtoStax SS Tracker desde el repositorio de GitHub, enlace del repositorio:

```
$ git clone
https://github.com/protostax/ProtoStax_ISS_Tracker.git
```

El código anterior instala la biblioteca de ePaper de Waveshare.

Paso 2. Comprobar si funciona la interfaz. Verificando que SPI esté realmente "activado" en su /boot/config.txt. Busque una entrada como la siguiente:

```
dtoverlay=spi=on
```

Paso 2.1 Comprobar que se puede visualizar el dispositivo SPI conectado.

```
$ ls /dev/spi*

/dev/spidev0.0 /dev/spidev0.1
```

está utilizando una versión anterior de python, deberá instalar python3

El código también depende de otras bibliotecas. Instalar las dependencias de spidev, RPi.gpio, Pillow y requests de la siguiente manera:

```
$ sudo apt-get install python3-spidev
$ sudo apt-get install rpi.gpio
$ sudo apt-get install python3-pil
$ sudo pip3 install requests
```

usted en el repositorio git anterior.

```
$ cd ProtoStax_ISS_Tracker
```

Ejecute el script de python iss.py (usando python3):

```
$ python3 iss.py
```

Si todos los pasos anteriores se han realizado correctamente, debería ver el mapa del mundo con la ubicación actual de la Estación Espacial Internacional en su pantalla de papel electrónico. Cada 30 segundos, la ubicación actual se actualiza.

Finalmente se puede visualizar de la siguiente manera, ver [8].

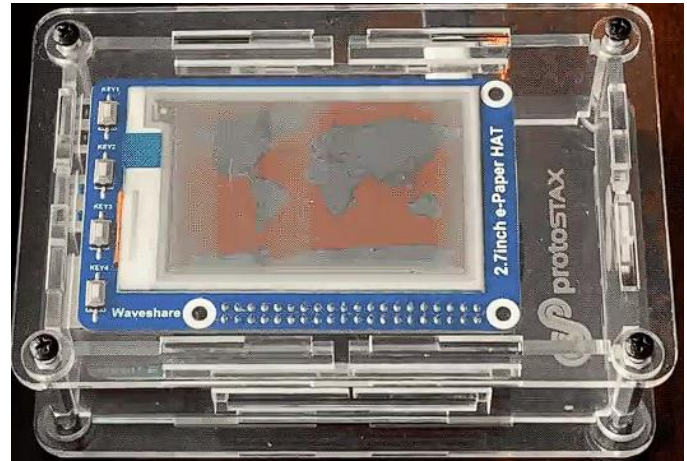


Fig. 11. ProtoStax ISS Tracker en acción

Proyecto 3

PC solar



Fig. 12. Pi solar

Descripción

El proyecto se trata de la creación de una estación de juegos portátil, especialmente pensada para mantener entretenidos a los niños de manera segura y ordenada. La motivación detrás de este proyecto fue evitar el uso de cables sueltos en presencia de niños, lo cual podría ser peligroso.

La solución consiste en una carcasa imprimible en 3D que alberga varios componentes clave: una pantalla táctil, dos altavoces, una Raspberry Pi 4, una batería tipo power bank y las conexiones necesarias para conectar la estación a un cargador solar. Esta configuración permite que la estación de juegos funcione de manera autónoma y portátil, sin la

necesidad de estar conectada a una toma de corriente constante.

Objetivo

Diseñar una estación de juegos portátil y segura mediante la integración de una Raspberry Pi 4 y una pantalla táctil en una carcasa impresa en 3D, permitiendo un entorno libre de cables.

Lista de materiales y su costo

TABLA 9 MATERIALES Y COSTOS		
Material		Costo
Raspberry pi 4	\$ 80.00	
Pantalla táctil de 7	\$ 109.00	
2 altavoces	\$ 10.00	
Amplificador estéreo clase AB Audio 3.7W	\$ 47.00	
Panel solar de 5W y 5V		
	\$ 22.90	
Powerbank de 10000 mAh	\$ 14.99	
4 mini imanes de neodimio	\$ 10.00	
Cables y adaptadores	\$ 15.00	
Piezas 3D	\$ 15.00	
Herramientas		
Soldador		
Pistola de pegamento	\$ 5.00	
Pegamento instantáneo	\$ 0.25	
Alicates	\$ 5.00	
Destornillador plano fino	\$ 2.00	

Guía de montaje y/o programación.

Paso 1. Pegar las piezas, ir uniendo dos mitades de la caja con el pegamento de impacto y los soportes.

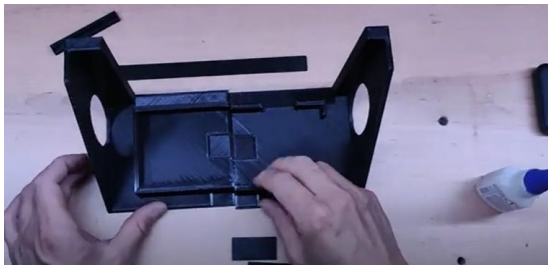


Fig. 13. Montaje de piezas

Paso 2. Se coloca la pantalla táctil

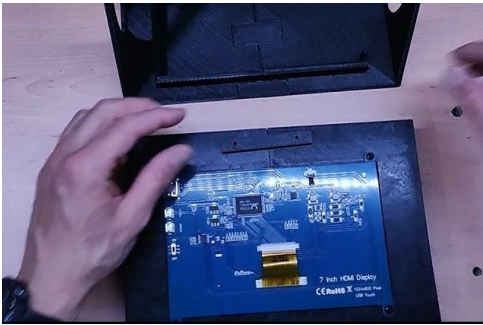


Fig. 14. Colocación de pantalla táctil

Paso 3. Colocar la Raspberry Pi en la parte superior de la pantalla, al mismo tiempo se debe colocar los cables USB y adaptador HDMI



Fig. 15. Raspberry Pi, USB y HDMI

Paso 4. Se colocan los imanes para fijar el panel solar.



Fig. 16. Imán y panel solar

Paso 5. Unir la pantalla táctil y la carcasa al mismo tiempo se terminar de realizar las conexiones HDMI.

Paso 6. Añadir los altavoces a la Raspberry Pi

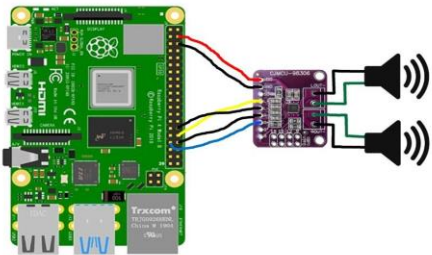
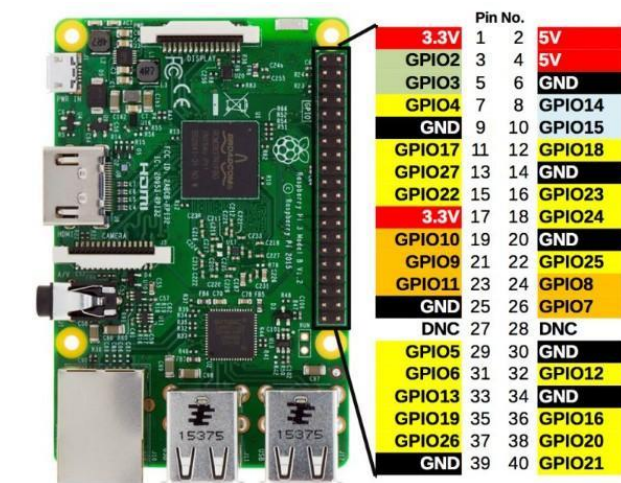


Fig 17. Conexión de altavoces



Pines Raspberry pi 4 (GPIO)	Pines amplificador
4 (5v)	VDD
6 (GND)	GND
30 (GND)	L-
32 (GPIO12)	L+
34 (GND)	R-
33 (GPIO13)	R+

Fig. 18. Pinout de conexión esquema

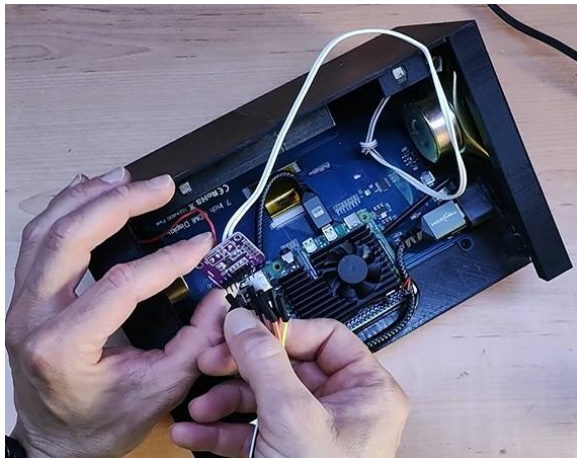


Fig 18.1. Conexión de altavoces

Los altavoces se fijan en la caja con pegamento caliente. Diríjase del esquema para conectar las conexiones entre los altavoces al amplificador estéreo y para los pines GPIO de la Raspberry. A continuación, esconde los cables y el amplificador a través de los agujeros de la caja para que quede bien organizado.

Para que los altavoces reproduzcan sonido a través de los pines GPIO de la Raspberry, tendrás que editar el archivo config.txt. Usando el sistema operativo Raspberry pi para este proyecto.

Paso 7. Conecte el banco de energía en la Raspberry Pi y enciéndalo.

Paso 8. Abra una terminal en el sistema operativo Raspberry Pi e ingrese:

```
sudo nano /boot/config.txt
```

Cuando el archivo este abierto tiene que ir hasta el final y añadir lo siguiente:

```
dtoverlay=audremap,pin_12_13
```

Guarde los cambios y salga del archivo.

Nos dirigiremos al altavoz, colocándolo en la parte superior derecha de la pantalla y con el botón derecho del ratón, seleccionaremos que la salida de audio sea a través del puerto AV jack.

Si todo salió bien, ahora deberías escuchar el sonido a través de los altavoces.

Paso 9. Pegue el soporte superior de la carcasa fijándolo con los tornillos y conectar el panel solar al powerbank, ver [9].



Fig. 19. Proyecto final PC solar

IV. CONCLUSIONES

El análisis exhaustivo de la tecnología Raspberry Pi ha revelado su potencial en la educación, la creación de prototipos y la automatización de proyectos. Desde sus inicios con modelos como la Raspberry Pi 1 hasta las poderosas capacidades del modelo 4, esta plataforma ha evolucionado constantemente, ofreciendo a los entusiastas de la tecnología una herramienta versátil y económica para explorar la informática y la programación.

Comparando Raspberry Pi con Arduino, se destacan las diferencias significativas en términos de funcionalidad y aplicaciones. Mientras Raspberry Pi se centra en enseñar informática y es ideal para proyectos de computación general,

Arduino está diseñado específicamente para la experimentación y la creación de proyectos de electrónica.

V. BIBLIOGRAFÍA

- [1] R. Pi, Getting started with your Raspberry Pi, 2024.
- [2] R. Pi, «Raspberry Pi 4 Tech Specs,» 2024. [En línea]. Available: <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/>. [Último acceso: 07 04 2024].
- [3] xataka, «Qué es Arduino, cómo funciona y qué puedes hacer con uno,» 23 Septiembre 2022. [En línea]. Available: <https://www.xataka.com/basics/que-arduino-como-funciona-que-puedes-hacer-uno>. [Último acceso: 07 04 2024].
- [4] Arduino, «¿Qué es Arduino?,» [En línea]. Available: <https://arduino.cl/que-es-arduino/>. [Último acceso: 08 04 2024].
- [5] J. Delgado, «Raspberry Pi vs. Arduino: diferencias y cuál es la mejor opción,» Computer, 23 Septiembre 2023. [En línea]. Available: <https://computerhoy.com/tecnologia/raspberry-pi-vs-arduino-diferencias-cual-mejor-opcion-1305172>. [Último acceso: 08 04 2024].
- [6] Y. Frenández, «Arduino y Raspberry Pi: qué son y cuáles son sus diferencias,» Xataka Basic, 28 Agosto 2018. [En línea]. Available: <https://www.xataka.com/basics/arduino-raspberry-pi-que-cuales-sus-diferencias>. [Último acceso: 08 04 2024].
- [7] Daniel, «Workday Progressbar With Google Calendar Integration,» AutoDesk Instructables, [En línea]. Available: <https://www.instructables.com/Workday-Progressbar-With-Google-Calendar-Integrati/>. [Último acceso: 08 04 2024].
- [8] S. Rajagopal, «International Space Station Tracker,» 23 Junio 202. [En línea]. Available: <https://www.hackster.io/sridhar-rajagopal/international-space-station-tracker-6afdc4>. [Último acceso: 08 04 2024].
- [9] A. Instructables, «Solar PC All-in-one Raspberry Pi,» [En línea]. Available: <https://www.instructables.com/Solar-PC-All-in-one-Raspberry-Pi/>. [Último acceso: 09 04 2024].